设计研究与应用

基于实时流技术的诈骗短信预警系统的设计与实现

王明

(北京邮电大学网络技术研究院 北京 100876)

摘 要: 设计并实现了诈骗短信预警系统。介绍了实时流处理技术以及该技术在诈骗短信预警系统中的应用。该系统以旁路监听的方式分流出到达短信中心的短信,并将其置于 flume 监听文件夹下面,kafka 缓存从 flume 端收集上来的信息,然后 storm 读取这些信息,并对这其进行过滤,统计分组,最后得到诈骗短信名单及可疑短信名单,并对诈骗短信的接收者发送预警信息。

关键词: 诈骗短信; 实时流处理; storm; flume

中图分类号: TP311 **文献标识码:** A **DOI:** 10.3969/j.issn.1003-6970.2015.01.007

本文著录格式: 王明. 基于实时流技术的诈骗短信预警系统的设计与实现[J]. 软件, 2015, 36(1): 32-37

The Design and Implementation of Real-time Streaming Technology of Early Warning System based on SMS Scams

WANG Ming

(*Institute of Network Technology, Beijing University of Posts and Telecommunications, Beijing* 1000876, *China*)

【Abstract】: Design and implement a fraud SMS warning system. Describes the real-time streaming technology and the application of the technology in the fraud SMS warning system. The system is a way to bypass monitoring shunt out SMS text messages to reach the center and placed flume monitor folder, kafka cached information collected from the flume end up, then storm to read the information, and that its filter statistical grouping, and finally get a list of fraud and suspicious SMS message list, and defraud the recipient to send SMS warning message[11].

【Key words】: SMS fraud; real-time stream processing; storm; flume.

0 引言

随着通信服务的不断扩展，利用移动通信服务传送诈骗短信的违法和不良信息的现象也随之出现并呈蔓延趋势，许多违法行为借助此方式发生，移动客户抱怨不断增多，投诉不断升级，社会各界也普遍关注.为了遏制诈骗短信带来的危害，本文提出基于实时流处理的诈骗短信拦截提示系统的解决方案，通过系统配置的超频门限和诈骗短信内容等过滤条件，实时产生诈骗短信告警信息，实现对诈骗短信的检测应用。

1 实时流技术介绍

本系统采用的是 flume+kafka+storm 的实时流处理框架。在这个框架中，flume 负责数据采集部分，实时监测文件夹下面文件的变化；kafka 充当了 flume 和 storm 之间数据缓冲的作用，用于对数据进行缓存。storm 是核心处理模块，用于对采集上来的数据实时的统计、分析过滤。下面将对该实时流处理技术里面用到的三个技术做详细的讲解。

1.1 Flume 介绍

Flume 是一个分布式、可靠、和高可用的海量日志聚合的系统，支持在系统中定制各类数据发送方，用于收集数据；同时，Flume 提供对数据进行简单处理，并写到各种数据接受方(可定制)的能力。Flume 的特点是可以通过手工配置，可以自动收集日志文件，在大数据处理及各种复杂的情况下，flume 经常被用来作为数据处

作者简介: 王明(1990–), 男, 山东人, 研究生在读, 研究方向: 大数据流式处理

理的工具，Flume 搜集日志的方式多种多样，比如可以检测文件夹的变化，可以监测端口信息等等，在本系统中采用的就是 flume 检测文件夹变化的方式来实时收集信息。本系统采用的 flume 版本是 flume-ng 1.4，架构如图 1 所示。

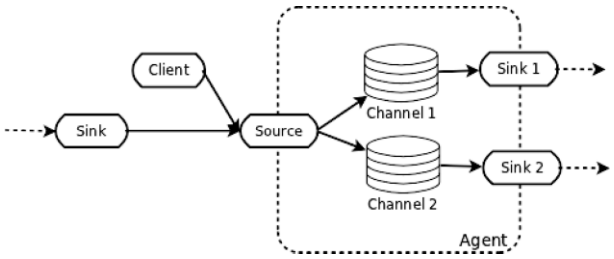

图 1　Flume-ng 架构图

在图 1 中 source：顾名思义，就是数据的来源，这个来源可以来自 Web Service 中封装的客户端(AVRO 客户端)，可以是 NETCAT 服务，也可以是一个不断增长的日志文件(tail -f)。

channel：提供了一层缓冲机制，来实现数据的事务性传输，最大限度保证数据的安全传输。这层缓冲可以在内存中，可以在文件中、数据库中，当然也可以是自定义的实现。

sink：将数据转发到目的地，或者继续将数据转发到另外一个 source，实现接力传输。可以通过 AVRO Sink 来实现。

在本系统中 source 使用的是对文件夹中文件变化进行监测的 Spooling Directory Source，channel 是用的内存空间，sink 是自定义的 kafkasink，用于向 kafka 发送数据。

1.2　Kafka 介绍

Kafka 是一个开源的分布式消息系统,由 LinkedIn 开发，主要用于处理 LinkedIn 的活跃数据。活跃数据主要包括页面浏览量(PV)、用户行为(登陆、浏览、搜索、分享、点击)、系统运行日志(CPU、内存、磁盘、进程、网络)等方面的数据。这些数据通常以日志的形式进行存储，然后周期性的对这些数据进行统计分析。

传统的日志分析系统主要用于处理离线数据，对于实时数据的处理，有较大的延迟性。现有的消息队列系统可以很好的用于日志分析系统对于实时数据的处理，但通常未被处理的数据不会写到磁盘上，这样不利于 Hadoop 这样的离线分析系统的使用。Kafka 可以很好的解决以上问题，能够很好的为离线和实时的分析系统提供服务。

kafka 是一种高吞吐量的分布式发布订阅消息系统， 它有如下特点：数据在磁盘上存取代价为 O(1)

一般数据在磁盘上是使用 BTree 存储的，存取代价为 O(lgn)；高吞吐率，即使在普通的节点上每秒钟也能处理成百上千的 message；显式分布式，即所有的 producer、broker 和 consumer 都会有多个，均为分布式的；支持数据并行加载到 Hadoop 中。Kafka 结构如图 2 所示。

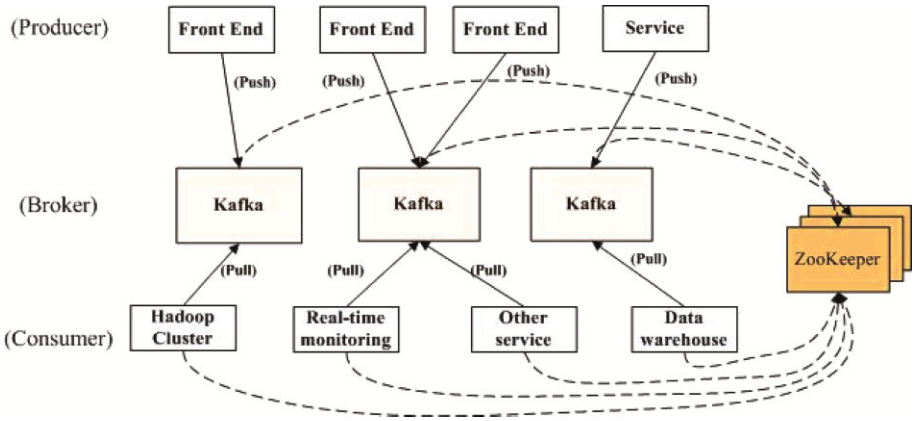

图 2　Kafka 结构图

在图 2 中 Producer 用于收集数据，Broker 用于数据的中间存储，consumer 用于数据的订阅，Kafka 是显式分布式架构，producer、broker(kafka)和 consumer 都可以有多个。Kafka 的作用类似于缓存，即活跃的数据和离线处理系统之间的缓存。多个 broker 协同合作，producer 和 consumer 部署在各个业务逻辑中被频繁的调用，三者通过 zookeeper 管理协调请求和转发。这样一个高性能的分布式消息发布与订阅系统就完成了。

1.3　Storm 介绍

Storm 是一个分布式的、容错的实时计算系统，遵循 Eclipse Public License 1.0,Storm 可以方便地在一个计

算机集群中编写与扩展复杂的实时计算，Storm 之于实时处理，就好比 Hadoop 之于批处理。Storm 保证每个消息都会得到处理，而且它很快——在一个小集群中，每秒可以处理数以百万计的消息[1]。可以使用任意编程语言来做开发。Storm 的优点如下：

- 简单的编程模型。类似于 MapReduce 降低了并行批处理复杂性[9]，Storm 降低了进行实时处理的复杂性。
- 服务化,一个服务框架,支持热部署,即时上线或下线 App。
- 可以使用各种编程语言。你可以在 Storm 之上使用各种编程语言。默认支持 Clojure、Java、Ruby 和 Python。要增加对其他语言的支持，只需实现一个简单的 Storm 通信协议即可。
- 容错性。Storm 会管理工作进程和节点的故障。
- 水平扩展。计算是在多个线程、进程和服务器之间并行进行的。
- 可靠的消息处理。Storm 保证每个消息至少能得到一次完整处理。任务失败时，它会负责从消息源重试消息。
- 快速。系统的设计保证了消息能得到快速的处理，使用 ZeroMQ 作为其底层消息队列。
- 本地模式。Storm 有一个“本地模式”，可以在处理过程中完全模拟 Storm 集群。这让你可以快速进行开发和单元测试。

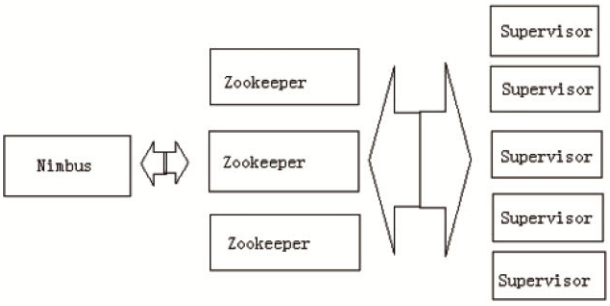

图 3　**Storm** 架构图

Storm 集群由一个主节点和多个工作节点组成。主节点运行了一个名为“Nimbus”的守护进程，用于分配代码、布置任务及故障检测。每个工作节点都运行了一个名为“Supervisor”的守护进程，用于监听工作，开始并终止工作进程。Nimbus 和 Supervisor 都能快速失败，而且是无状态的，这样一来它们就变得十分健壮，两者的协调工作是由 Zookeeper 来完成的。ZooKeeper 用于管理集群中的不同组件，ZeroMQ 是内部消息系统，JZMQ 是 ZeroMQMQ 的 Java Binding。有个名为 storm-deploy 的子项目，可以在 AWS 上一键部署 Storm 集群，如图 3 所示。

2　诈骗短信预警系统介绍

2.1　业务简介

诈骗短信通常有如下特点：用一个主叫号码在一段时间之内向大批量的其他被叫号码发送诈骗信息；短信内容重复且含有一些敏感词汇，像“银行”，“汇款”等。针对上面诈骗短信的特点，该预警系统通过配置关键词策略，将符合诈骗关键词策略的短信筛选出来，通过设置预警阈值，将短信内容相同的短信进行统计，并和阈值进行比较，大于阈值则将其放入可疑名单短信名单中[2]。

2.2　业务流程介绍

业务流程的介绍主要是收集到 storm 中的短信在 storm 中进行处理的一个过程，流程如图 4 所示：

- Storm 将收集上来的信令信息进行清洗，筛选出有用的信息，然后将信息下发。
- 统计在时间段 t(现阶段 t 定义为 5s)内，一共有多少条消息下发，现单机环境下 5s 内大约有 3 万条短信息下发到下面的流程中。
- 过滤白黑名单。白名单就是一些官方的主机号码，例如 10086,10010 或者银行类的官方号码，将这些短信剔除，可以提高系统效率。黑名单可以手动录入，也可以通过前面的判断在系统运行期间不断增加，当遇到主叫号码是黑名单里面的号码时，就向被叫号码发送一条预警短信。
- 统计 t 时间段内，相同短信内容及相同主叫号码的短信数目。
- 将 4 步骤中统计出来的短信信息与诈骗短信策略关键字进行对比，如果与其中的某一个诈骗短信策略相同，则将该短信号码置为诈骗短信，并跟据这条短信符合的诈骗短信策略选择合适的回复短信，给这条短信的被叫发送带有相应回复短信内容的预警短信。
- 对 5 步骤中不符合诈骗短信策略的短信根据其数量与系统的阈值进行比较，判断其是否是可疑短信，

如果相同短信的数目大于阈值，则将这类短信类型置为可疑短信，并根据其数目的多少，对其可疑程度进行判断，设置可疑级别为高、中、低，若短信数目小于阈值，过滤不处理。

- 对经过步骤 4 流出的短信，统计其 t 时间内诈骗短信及可疑短信的数目，并与 t 时间内总短信数目汇总，在 web 端进行实时展示。
- 将可疑短信入库，在 web 端进行展示，对可疑程度高的短信进行人工判断是否属于诈骗短信，如果属于诈骗短信，则将其关键字入库，并将号码放入黑名单，并给被叫在 web 端发送预警提示短信。
- t 时间段内结束，对下一个 t 时间段短信进行统计分析。

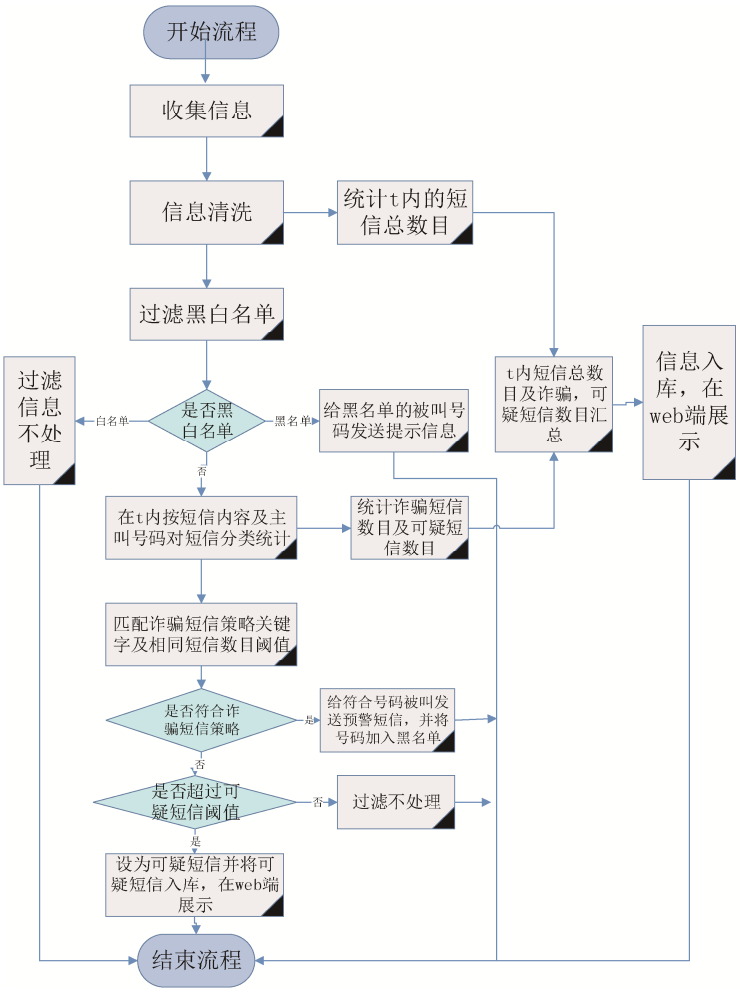

图 4　业务流程图

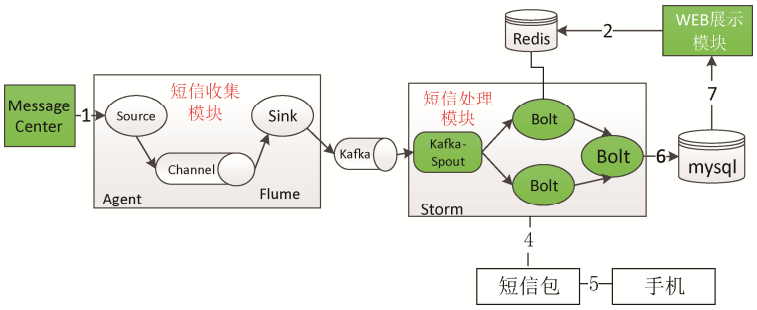

图 5　系统架构图

3　诈骗短信预警系统设计

3.1　整体设计框架

诈骗短信提醒具有实时性，即被叫在收到诈骗短信之后系统能够立即发现并进行短信提示，因此在实现时采用 flume+kafka+ storm 实时流处理框架比较合适[3]。Flume 用于实时采集信息内容；kafka 提供一个消 息缓冲队列，防止数据的采集端(flume)及处理端(storm)处理速度不一致时，数据的丢失；storm 是一个分布式实时流处理框架，对数据进行分布式实时处理。基本架构如图 5 所示。

3.2　短信收集模块

短信收集模块是指收集从 Message Center 中分离出的短信。用分光器将流向短信中心的数据分流出来，然后将数据放到服务器下面的文件夹下面，用 flume 实时监测文件的增长，获得实时包含短信的数据。

短信收集模块的实现主要是通过 Flume 来实现的，Flume 提供了用于监听文件变化的 Spooling Directory Source，通过将配置文件中的 Source 类型设置为 spooldir，并对相关参数进行设置即可对指定目录内的文件进行实时监测。

Channel 选用的是 memory channel,作用是缓存从 source 端收集上来的消息，选用内存作为 channel 的作用是读取速度快，保证实时性，sink 端用于从 channel 读取信息，这里需要将检测到的新增记录存储到 kafka 缓冲队列中，自定义 Flume 的 Sink，通过在 Flume 的工程中引入 Kafka 的 API 来自定义 Kafka Sink。

3.3　短信实时处理模块

短信实时处理模块是诈骗短信预警系统的核心模块，实现是用 Storm 实时处理框架，通过统计时间 t 内的相同短信数目及匹配短信关键字来找出可疑短信、诈骗短信。通过设计 storm 的 topology 来实现对输入流的处

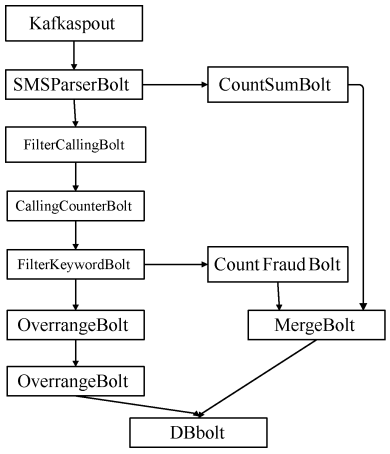

图 6　**Topology** 设计图

理，在 topology 中，流是一个不间断的无界的连续 tuple，通过在 topology 各个 bolt 里面不断的处理，将信息进行筛选过滤[4]，实时处理模块的 Storm topology 图如图 6 所示。

Kafkaspout 用于从 kafka 消息队列读取数据，将数据读入 storm 里面，原封不动的发送给解析 SMSParserBolt，在该 bolt 里面将数据进行清洗，保留有用的信息，然后将消息按 Shuffle Grouping 下发给 FilterCallingBolt ，该 Bolt 用于过滤黑白名单，FilterCallingBolt 根据短信内容和主叫号码将 tuple 按照 Fields Grouping 下发给 CallingCounterBolt，在该 Bolt 里面统计时间 t 内拥有相同短信内容和主叫号码的短信数目，然后将统计后的结果按照 Fields Grouping [5]下发到 FilterKeywordBolt 进行关键词匹配，匹配成功则判定该主叫号码发送的短信为诈骗短信，并给该主叫发送预警短信，若匹配不成功则将该短信下发到 OverRangeBolt 进行数目判定，若超过判定阈值，则将该短信置为可疑短信，然后下发到 DBbolt 进行入库操作，CountSumBolt 和 CountFraundBolt 用于统计 t 时间内的短信总数目和诈骗短信数目，然后在 MegeBolt 里面进行合并后，下发到 DBbolt 里面进行入库操作[7]。

3.4　预警短信发送模块

预警短信发送模块是指当发现主叫号码发送的短信是诈骗短信的时候，会给被叫用户发送一条预警提示短信，防止被叫用户收到诈骗短信后上当受骗。

预警短信的发送在程序的三个地方可能被触发，第一个地方是过滤黑白名单时，若主叫号码在黑名单，会立即给主叫号码对应的被叫号码发送预警短信。第二个地方是 FilterKeywordBolt 里面进行关键字配置，若短信内容符合诈骗短信关键字策略，则会给主叫号码发送对应诈骗短信关键字策略的相应预警短信。第三个地方是在 web 页面上当判定可疑短信为诈骗短信的时候，可以点击页面上的发送预警短信按键，在后台给被叫用户发送预警短信。

3.5　web 展示模块

Web 展示模块用于向用户直观的展示统计信息及各个功能模块[10]，主要包括如下方面：

- 根据设定好的显示粒度(秒/分钟/小时)实时显示诈骗短信疑似号码数量、归属地、诈骗短信发送量等信息。
- 对短信关键字进行增删改查。
- 白名单管理。对白名单进行增删改查，白名单中的号码不在监测范围内。
- 黑名单管理。对黑名单中被确认的诈骗号码进行增删改查。
- 阈值设置。对可疑短信阈值进行设置。
- 记录查询。根据主叫号码提供对短信类型(垃圾短息，诈骗短息)、策略、发送数量、疑似度、归属地、开始时间、结束时间、短信内容的查询。
- 记录统计。主叫号码短信(垃圾短信、诈骗短信)统计，根据不同的时间段进行统计短信量,主叫号码为空时，查询所有的号码，主叫号码不为空时，匹配输入的主叫号码。

3.6　数据库模块

本系统用到的数据库包括 redis 和 mysql 数据库，redis 数据库读写速度快，用于存储在程序中经常用到的数据。Mysql 数据库用于存储 web 展示相关的数据。

黑白名单，统计的短信数目及诈骗短信数目，关键字策略存储在 redis 数据库里面，存储的类型为 String。关键字策略、诈骗短信统计记录、被叫号码统计记录存储在 Mysql 里面，表结构见表 1~3。

表 1　关键字策略

字段名称	类型	说明
ployId	int	策略 ID
ployName	string	策略名称
smsType	int	短息类型 0-垃圾短息，1-诈骗短息
keywords	string	关键字(多个用‘,’号隔开)
remindContent	string	提醒内容模板

4　业务系统方案分析

随着通信技术的发展，诈骗短信层出不穷，诈骗短信的手段也在不断提高，对社会造成的危害也在不断加剧，但现在对诈骗短信进行预防的技术少之又少，市场上现有的预防诈骗短信技术有 360 手机安全卫士，腾讯手机助手等借助手机终端进行预防诈骗短信的方法[6]。相对于这种预防手段，基于实时流

表 2　被叫号码记录

字段名称	类型	说明
callingNumber	string	主叫号码
calledNumber	String	被叫号码
smsContent	string	短息发送内容
time	timestamp	发送时间
smssend	int	疑似短信 0-未发送 1-已发送
sendtime	timestamp	提醒短信发送时间

表 3　诈骗短信统计记录

字段名称	类型	说明
callingNumber	string	主叫号码
smsCount	int	短息数
smsSort	int	短息类型 0-垃圾短息，1-诈骗短息
smsContent	string	短息提醒内容
ployId	int	策略 ID
startTime	timestamp	开始时间
endTime	timestamp	结束时间
source	string	号码归属地
rank	int	疑似度(高、中、低)
smsType	int	0-MO 1-MT

处理技术的诈骗短信预警系统具有如下几个方面的优势：

- 对手机终端没有任何要求。借助于手机软件如 360 手机助手等进行诈骗短信预防的方法，对手机终端操作系统要求较高，在不能安装非智能系统的终端上这种方法便没有了用武之地。而本系统对手机终端没有任何要求，只要能够接收短信即可实现。
- 能够智能判断收到的短信是否是诈骗短信。据调研手机软件的判断方法大多是通过判定诈骗短信的主叫号码是否在诈骗短信黑名单中，而诈骗短信黑名单的获得是通过其他用户的举报。这样对一些尚未出现在诈骗短信黑名单中的诈骗短信便失去了判定方法，更重要的是，如果更换了主叫号码进行诈骗短信，则手机软件端便无法根据主叫号码进行判断。而本系统可以根据诈骗短信关键词进行判断，能够智能的判断收到的短信是不是诈骗短信，而且也排除了主叫号码的干扰，只通过短信内容进行判断。
- 能够及时的发现市场上新出现的诈骗短信并进行预防。对于新出现的诈骗短信，手机软件的反应会很慢。对于新出现的诈骗短信的特点是发送量比较大，针对这种特点，本系统会统计相同短信的数目，当数目超过设定的阈值的时候，就会标记为可疑短信，然后在 web 端进行预警提示，通过人工进行判断，若判断出是诈骗短信，即可以给被叫发送预警短信，并将主叫号码加入黑名单，设置相应的诈骗短信关键字。
- 能够根据不同的诈骗短信内容发送不同的预警短信信息。根据短信关键字判断的诈骗短信，可以发送对应的预警短信，起到更好的诈骗短信预防方法。

本论文中的业务系统虽然有很多优势，但并非完美，仍然存在可以改价的地方。一方面，当数据量很大时，处理时延可能会增大，实时性受到一定程度的影响，可以通过对系统优化，尽量减少系统的处理时延；另一方面，诈骗短信关键词过滤不智能，诈骗短信关键词策略，现只能通过手动添加的方式，效率差，后期可以引入人工智能的方法，例如采用朴素贝叶斯分类方法进行关键词匹配及更新关键词库[8]。

5　总结

本文介绍了一种基于实时流处理技术的诈骗短信预警系统。系统实现了被叫用户收到诈骗短信时，系统会及时给被叫发送预警信息，防止被叫上当受骗的功能。遏制了诈骗短信的猖獗，保护了运营商用户的利益。对电信运营商在信息安全方面提供了新方法。移动通信技术在不断的发展，安全方面的问题也在不断加剧，移动安全变的越来越重要。相信本文分析设计的业务系统具有较好的应用和发展前景，也必将对电信运营商移动安全方面的研究提供帮助。

参考文献

[1] 王鹏. 一种基于Storm编程模型的迭代Topology方案. 成都信息工程学院学报, 2014(15).

[2] Jingling Zhao. Implementation of K-Means Based on Improved Storm Model. 国际会议, 2013(11).

[3] 李继刚. 短信自动分类技术研究与实现. 东华大学. 2012.11.06

[4] 坚兆文. 基于MTK平台的手机垃圾短信过滤系统设计与实现. 中南大学, 2012(04).

[5] 薛冰. 基于Android系统的主动拦截技术的研究. 北京邮电大学, 2013(12).

[6] 李凡. 基于内容的短信智能分类系统的设计与实现. 东北大学, 2010(06).

[7] 刘庆瑜. 基于决策树分类的手机垃圾短信过滤器的设计与实现. 浙江工业大学, 2011(08).

[8] 黄文良, 陈纯, 罗云彬. 一种高效垃圾短信过滤系统的实现. 电信科学. 2010.3.17

[9] 白亚鲁. 云计算环境下大规模数据处理的研究[J]. 软件, 2013, 34(5): 128–129.

[10] 张慧宁. 基于web技术的实验室开放管理系统设计[J]. 软件, 2013, 34(11): 52–54.

[11] XIE Sheng, Chen Hang, Yu Ping, et al.High-Accuracy Frequency Estimator Based On Average Amplitude Spectrum[J]. The Journal of New Industrialization, 2011, 1(3): 31–36.